

# Kaba Kod (Pseudo Code)

- Programcılar sıklıkla algoritmaları, yalnızca insan gözlerine yönelik bir şekilde tanımlamak isterler.
- Bu tanımlamalar bilgisayar programı ya da bir akış şeması değildir.
- Daha çok yapılandırılmış bir metin tanımıdır.

# Kaba Kod (Pseudo Code)

- Bu tanımlamalar veri yapısının ya da algoritmanın **üst düzey analizini** kolaylaştırır.
- Bu tanımlamalar sözde kod (pseudo code) olarak adlandırılmaktadır.

# Kaba Kod (Pseudo Code)

- Sözde kodun, eşdeğer bir gerçek yazılım kodu parçasının olacağından **daha küçük** olduğuna dikkat edin.
- Ek olarak, sözde kodun okunması ve anlaşılması kolaydır.

# ekranaYaz (n)

- Birden  $n$ 'e kadar olan sayıları ekrana yazdıran algoritmanın kaba kodlarını yazalım:

Algoritma *ekranaYaz*( $n$ )

**Giriş:** Limit değerini temsil eden  $n$  sayısı.

**Çıkış:** Yok

$n$  değerini oku

$i \leftarrow 0$

**while**  $i \leq n$  **do**

$i$  ekrana yaz

**end while**

# ikiUzeri(n)

- İkinin kuvveti alan bir algoritmanın kaba kodunu oluşturalım.  $n$  değeri dışardan girilecek ve  $2^n$  şeklinde hesaplama yapılacaktır.

Algoritma *ikiUzeri(n)*

**Giriş:** Üst değerini temsil eden  $n$  sayısı.

**Çıkış:** İki üzeri  $n$

$n$  değerini oku

$üst \leftarrow 1$

**Tekrar et  $n$  defa**

$üst \leftarrow üst * 2$

**tekrarı durdur**

# ardisikSayilarinToplami (n)

- Limit olarak klavyeden belirtilen bir  $n$  sayısına kadar olan ardışık sayıların toplamı için kaba kod tasarımı yapınız.

Algoritma *ardisikSayilarinToplami (n)*

**Giriş:** Limit değerini temsil eden  $n$  sayısı.

**Çıkış:** Toplam sonucu

$n$  değerini oku

$i \leftarrow 0$

$toplama \leftarrow 0$

**while**  $i \leq n$  **do**

$toplama \leftarrow toplama + i$

$i = i + 1$

**end while**

Toplamı yaz

# arrayMax

- **N** elemandan oluşan bir dizideki en büyük elemanı bulan bir problem üzerinde duralım.
- Bu sorunu çözmek için, **for** döngüsünü kullanarak **A**'nın elemanlarını tarayan **arrayMax** adlı bir algoritma kullanabiliriz.



# arrayMax

- arrayMax fonksiyonuna ait kaba kod tanımı aşağıdaki gibi yapılır.

**Algorithm** arrayMax( $A, n$ ):

***Input:*** An array  $A$  storing  $n \geq 1$  integers.

***Output:*** The maximum element in  $A$ .

$currentMax \leftarrow A[0]$

**for**  $i \leftarrow 1$  **to**  $n - 1$  **do**

**if**  $currentMax < A[i]$  **then**

$currentMax \leftarrow A[i]$

**return**  $currentMax$

# arrayMax

- Kaba kod yardımıyla arrayMax adlı algoritmanın doğruluğu üzerinde basit bir tartışma gerçekleştirebiliriz.

**Algorithm** arrayMax( $A, n$ ):

*Input:* An array  $A$  storing  $n \geq 1$  integers.

*Output:* The maximum element in  $A$ .

$currentMax \leftarrow A[0]$

**for**  $i \leftarrow 1$  **to**  $n - 1$  **do**

**if**  $currentMax < A[i]$  **then**

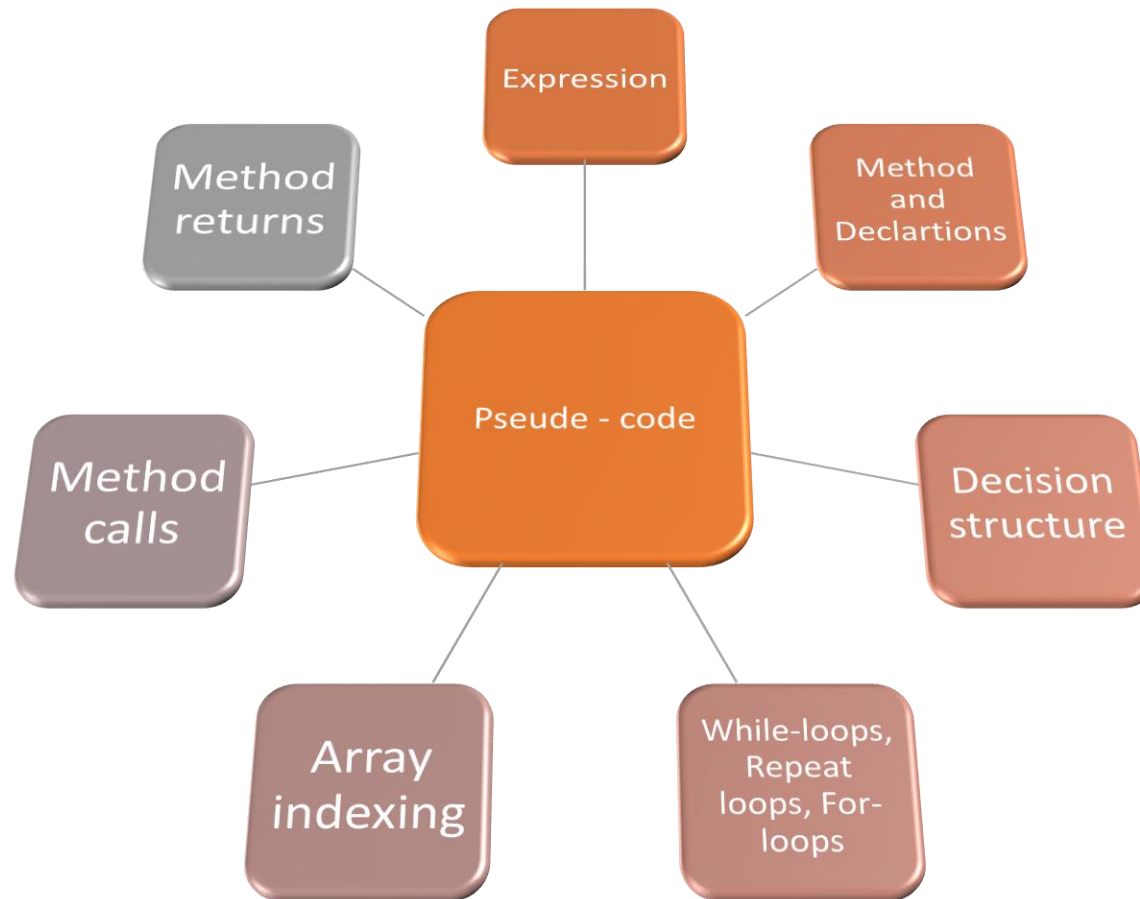
$currentMax \leftarrow A[i]$

**return**  $currentMax$

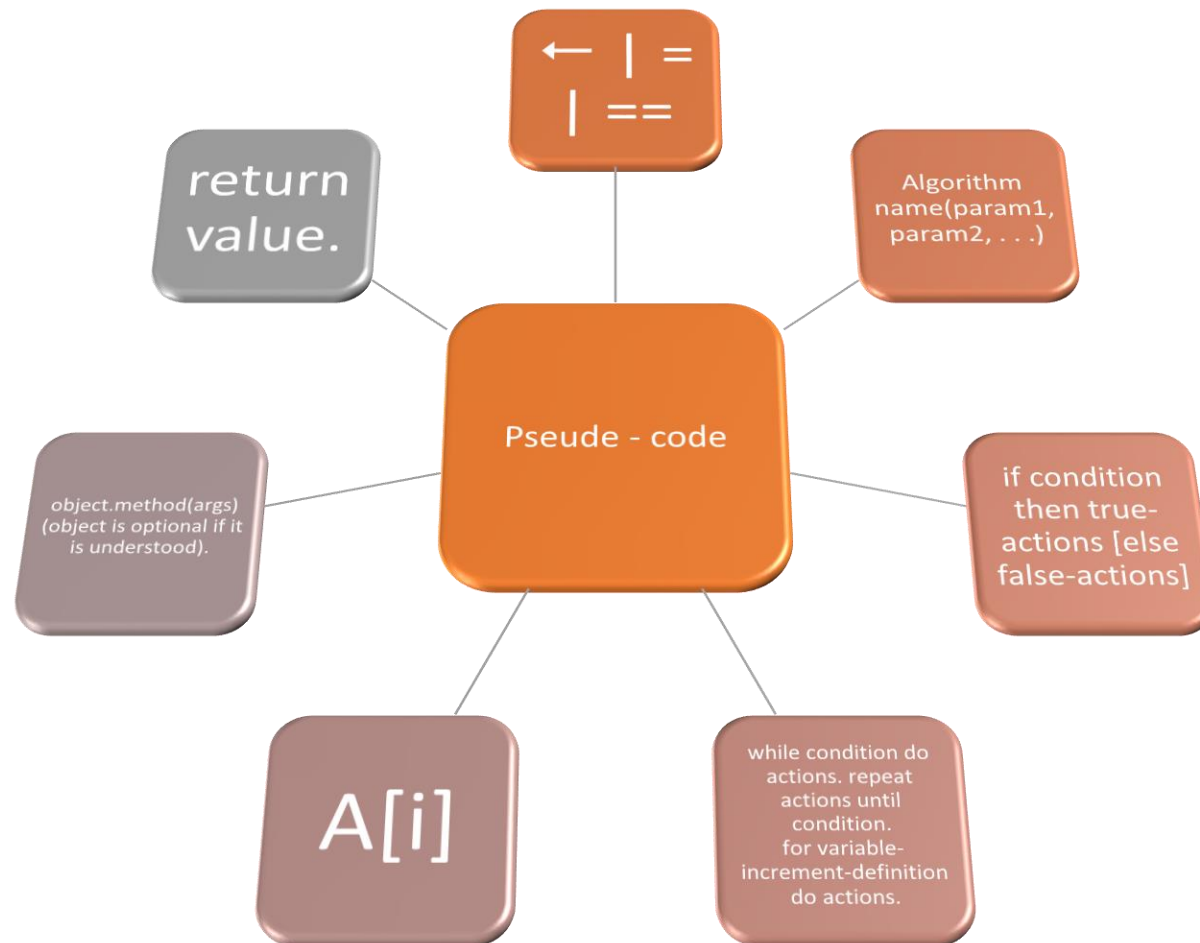
# arrayMax

- `currentMax` değişkeni dizinin ilk elemanını tutarak, ilk elemanın dizideki maksimumu değeri tuttuğunu varsaymaktadır.
- Daha sonra  $n-1$  iterasyonda dizinin her bir elemanı `currentMax` ile karşılaştırılıp, dizide daha büyük bir eleman bulunması halinde güncellenmektedir.

# Kaba Kod (Pseudo Code)

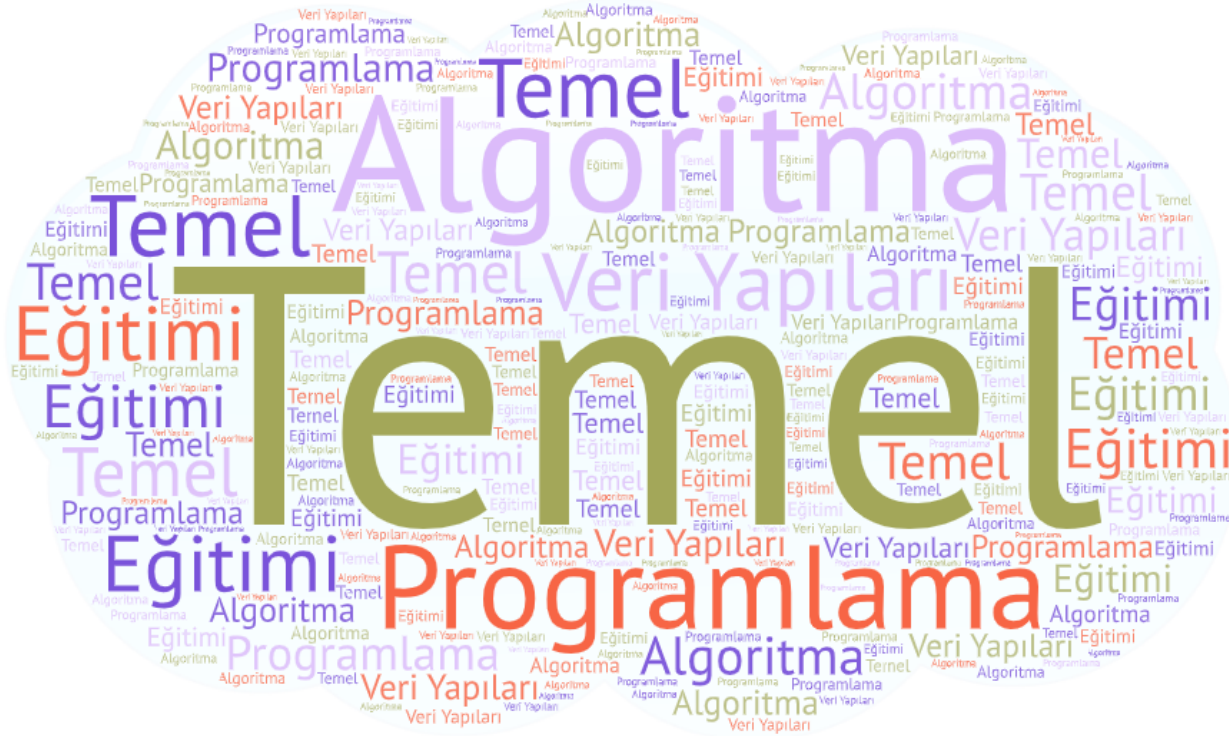


# Kaba Kod (Pseudo Code)



# Kaba Kod (Pseudo Code)

- Bir kaba kod yazarken bu kodu makinelere değil, insanların okuyacağı unutulmamalıdır.
- Bu nedenle **anlaşılabilirlik yüksek seviyede tutulmalı**, teknik detaylardan kaçınılmalıdır.
- Kaba kod algoritmaları tanımlamak için **yüksek seviyeli bir yoldur** ve kaba kod üzerinden algoritma analizi yapılabilmektedir.



**KABA KOD**